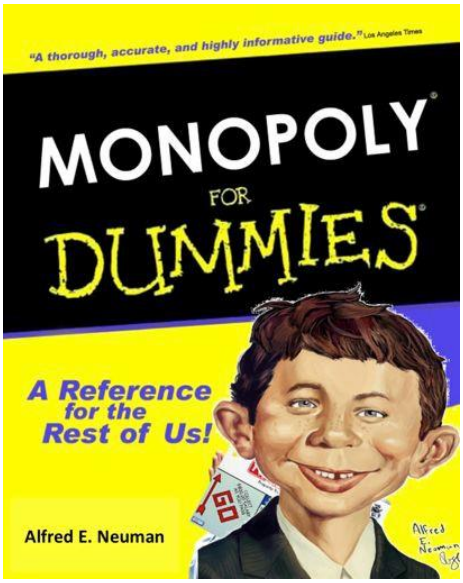
CS 2336 – Lab 3 Assignment

Monopoly for Dummies (100 points)



We have found a new home for the newly retired Monopoly *Iron* game piece. A new *very* simple version of the game we'll call *Monopoly for Dummies.*

You will be building a partially completed object> oriented Java version of this game in Lab 5, and will build additional functionality in Lab 6.

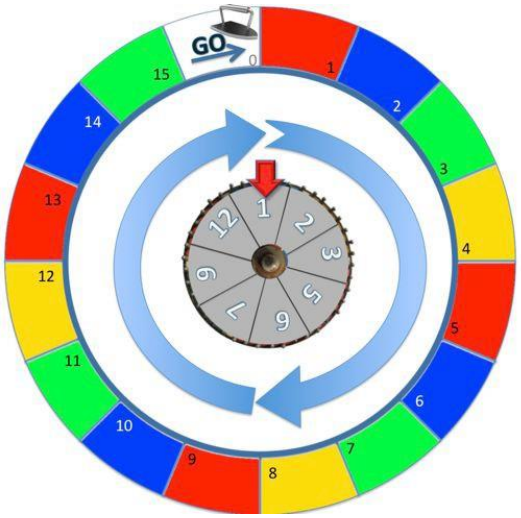
Along with new code, you should also plan to reuse the *Spinning Wheel* code that you developed in lab 4

NOTES:
Each program should include comments that explain what each block of code is doing. Additionally, the programs should compile without errors, and run with the results described in the exercise. The following deductions will be made from each exercise if any of the following is incorrect or missing:

- Proper formatting [5 points]
- Proper names for classes and variables [5 points]
- Comments [5 point]
- Program doesn't compile [10 points]
- Source code (java file) missing [10 points]
- Executable (class file) missing [10 points]
- Missing array where an array was required [5 points]
- Missing loop where a loop was required [5 points]
- Missing class from the design provided [10 points]
- Missing method from the design provided [5 points[

The Game (version 1.0)

This simple game is comprised of a circular game board comprised of 16 colored squares. Your game token is the *iron*, which starts on the GO square.



Game play begins when the player spins a wheel which contains the numbers 1, 2, 3, 5, 6, 7, 9, and 12.

The *iron* will then move ahead the number of squares based on the number that the spinner's arrow points to after the spinner stops.

The order of the colored squares is:

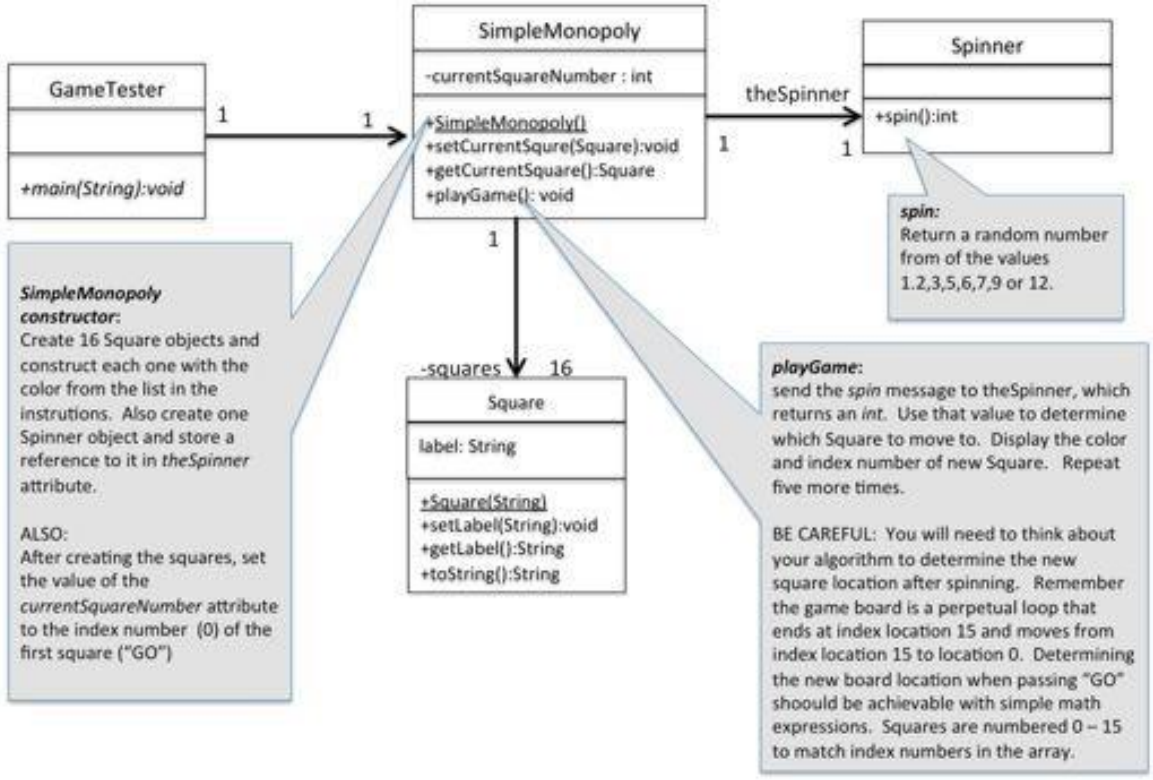
- White (GO)
- Red
- Blue
- Green
- Yellow
- Red
- Blue
- Green
- Yellow
- Red
- Blue
- Green
- Yellow
- Red
- Blue
- Green

After landing on the new square, display the *iron*'s new location. Repeat for **ten** more spins, displaying the new location after each spin.

That's it! This first version is not very exciting, but it will allow you to build the basic structure of the game that we can build upon.

Use the design on the following page as your specification for which classes to build, and which methods and attributes belong in each of those classes. CAUTION: You will have to be clever when figuring out how to navigate the board as you move from the last square back to the first square within the array of Square objects.

SimpleMonopoly Design:



GameTester code:

```
public class GameTester {
public static void main(String[] args) {
 SimpleMonopoly sm = new SimpleMonopoly();
 sm.playGame();
}//end main
}//end class GameTester
```

Sample Output:

```
> java GameTester
Iron is starting on GO square #0
Spun 9...Iron is now sitting on Red square #9
Spun 9...Iron is now sitting on Blue square #2
Spun 1...Iron is now sitting on Green square #3
Spun 12...Iron is now sitting on Green square #15
Spun 6...Iron is now sitting on Red square #5
Spun 9...Iron is now sitting on Blue square #14
Spun 7...Iron is now sitting on Red square #5
Spun 12...Iron is now sitting on Red square #1
Spun 6...Iron is now sitting on Green square #7
Spun 5...Iron is now sitting on Yellow square #12
```

These solutions are due by 6:00AM on Wednesday, October 29